CHAPTER 4
PHASE-SPACE-CLUSTERING BASED RECOMMENDER SYSTEMS FOR STOCK TRADING

4.1 Introduction
An alternative approach to design of stock trading recommendations by integrating the concepts from chaos theory and clustering is presented in this chapter. The theory of dynamical systems suggests that signals exhibiting chaotic behavior in time domain can exhibit deterministic behavior in their phase space. Hence, instead of directly employing the stock price time series data, as done in chapter 3, an attempt has been made in this chapter to utilize the phase space representation of the stock price time series data, along with clustering techniques and ARTMAP neural network to generate stock trading recommendations.

In the present study, phase-space representation of the stock price time series under consideration is first obtained with the help of the optimal lag value \( L_p \), obtained as the lag at which the first minimum of mutual information is obtained, and the embedding dimension \( M_p \), obtained using the false nearest neighbor technique. The points in the phase space are then grouped into similar clusters using clustering techniques. ‘Profitable’ clusters are identified from the obtained clusters and the points belonging to these clusters are used as training samples to the ARTMAP neural network. The trained network is then used to generate ‘Trade’/ ‘No Trade’ recommendations. Performance of the proposed system is validated on nine different stocks drawn from three different stock markets- US, UK and India. The US and UK represent mature markets and India is chosen as the representative of emerging markets. An attempt has also been made to select the stocks from different sectors/industries to avoid any sector specific bias. Three different variants of the proposed system are evaluated on eight different performance measures. Performance is also compared to three other recommender systems and the traditional benchmark: B&H strategy.
Rest of the chapter is organized as follows: section 4.2 presents the detailed description of the proposed system and its components. Section 4.3 describes the three recommender systems to which the performance of the proposed recommender is compared. Section 4.4 presents the results and the conclusions are presented in section 4.5.

4.2 System description

The system proposed in the present study takes a novel approach to generating stock trading recommendations by integrating concepts from chaos theory and data mining. The stock price time series tend to exhibit chaotic behavior, making the task of forecasting such time series a difficult proposition. It is known that a time series that exhibits chaotic dynamics in time domain tends to exhibit deterministic dynamics in its phase space representation. In the present study, the stock price time series is first converted to its phase space representation. Assuming the training set to be a 1×N vector of daily stock prices, phase space representation is obtained by finding the lag and the embedding dimension from the time series. The optimal value of lag ($L_p$) is identified by finding the first minimum of mutual information. The optimal embedding dimension ($M_p$) is then obtained with the help of false nearest neighbor technique. Using $L_p$ and $M_p$, the phase space representation of the time series can be represented as an $N-(M_p-1)L_p \times M_p$ matrix with each row of the matrix representing a single $M_p$ dimensional point in its phase space. As the next step, the $N-(M_p-1)L_p$ points are clustered together into similar groups (in the present study, three different clustering techniques are evaluated: the k-means, the Unweighted Pair Group Method with Arithmetic Mean (UPGMA) clustering and the Fuzzy C-Means (FCM) clustering). A detailed description of the clustering techniques employed and the parameters used in the clustering process is presented in section 4.2.2. Optimal number of clusters is identified with the help of silhouettes. Once the points are clustered into, say $G$ number of clusters, the ‘profitable’ clusters are identified from the cluster centroids. The process is described in section 4.2.4. Assuming that out of $G$ clusters, $P$ clusters ($P \subset G$) are found to be ‘profitable’. The points belonging to these $P$ clusters along with their complement coded vectors form the input to the ARTMAP network along with the ‘Trade’ (+1) or ‘No Trade’(-1) recommendations as the corresponding target. The ARTMAP maps the relationship between the inputs and the trading recommendations. ARTMAP working and the parameters used in the present study are described in section 4.2.5. Once the
network has been trained, the system can be used to generate trading recommendations. The proposed system block diagram is presented in Figure 4.1.

![Proposed clustering –ARTMAP based trading recommender block diagram](image)

**Figure 4.1 Proposed clustering –ARTMAP based trading recommender block diagram**

Performance of the proposed recommender system is compared to four other systems, namely, the phase space representation-clustering based recommender (three variants based on k-means, UPGMA and FCM are evaluated), phase space representation-ARTMAP based recommender, ARTMAP based recommender trained with the original time series and the traditional benchmark B&H strategy. Each of the systems is evaluated on eight different performance measures, as suggested in (Brabazon and O’Neill, 2006), namely, TP, AP, P/ST, L/LT, MD, TT, PF and WR.
The proposed system is validated on nine stocks drawn from three different stock markets. The markets considered are BSE - India, the LSE - UK and the NYSE - USA. The stock markets considered were chosen so as to represent both the categories of markets: the Indian stock market is considered to be an emerging market and the UK is considered to be a more mature market. The US market NYSE is the largest by both market capitalization and the trading volumes and hence stocks from NYSE are included as well. While selecting the specific stocks, an attempt is made to include stocks from diverse sectors so that the results do not reflect any sector-specific bias. The stocks considered are:

India: Cipla, HUL, RIL
UK: RBS, GSK, SBY
USA: Boeing, 3M, Yahoo.

First 80% of the data for each stock is used for training and the remainder 20% is used for testing. The time frames considered are presented in Table 4.1.

<table>
<thead>
<tr>
<th>Market</th>
<th>Stock</th>
<th>Time Frame</th>
<th>No. of Days</th>
</tr>
</thead>
<tbody>
<tr>
<td>India</td>
<td>Cipla</td>
<td>28/9/2007-6/10/2008</td>
<td>254</td>
</tr>
<tr>
<td></td>
<td>RIL</td>
<td>01/01/2010-24/2/2011</td>
<td>291</td>
</tr>
<tr>
<td>UK</td>
<td>GSK</td>
<td>26/12/2002-11/11/2003</td>
<td>229</td>
</tr>
<tr>
<td></td>
<td>RBS</td>
<td>28/2/1995-23/5/1996</td>
<td>323</td>
</tr>
<tr>
<td></td>
<td>SBY</td>
<td>01/01/2009-31/12/2009</td>
<td>260</td>
</tr>
<tr>
<td>US</td>
<td>Boeing</td>
<td>04/1/2010-31/12/2010</td>
<td>252</td>
</tr>
<tr>
<td></td>
<td>3M</td>
<td>03/1/2011-13/4/2012</td>
<td>323</td>
</tr>
<tr>
<td></td>
<td>Yahoo</td>
<td>04/1/2011-10/5/2011</td>
<td>341</td>
</tr>
</tbody>
</table>

4.2.1 Selection of the inputs

It is believed that the stock price time series data tend to be chaotic in nature (LeBaron, 1994), (Chen, 1996) and (Loffredo, 1999). It has also been observed in (Brockwell and Davis, 1987) that chaotic time series exhibit deterministic behavior in phase space. Hence in this study, as the first step, the stock price time series is converted to its phase space representation. The process of generating phase space representation from the time series data is explained as follows:

Consider a time series consisting of \( N \) samples defined as a \( I \times N \) vector
Then according to (Zhang et al., 2005), the phase space representation of $Y$ is in the form of an $M_p \times N-(M_p-1)L_p$ dimensional embedded vector $Y_E$, defined as

$$Y = \{ y(1), y(2), \ldots, y(N) \} \quad (4.1)$$

$$Y_E = \begin{bmatrix}
   y(1 + (M_p - 1)L_p) & \ldots & y(1 + L_p) & y(1) \\
   y(2 + (M_p - 1)L_p) & \ldots & y(2 + L_p) & y(2) \\
   \vdots & \ldots & \vdots & \vdots \\
   y(N) & \ldots & y(N - (M_p - 2)L_p) & y(N - (M_p - 1)L_p)
\end{bmatrix} \quad (4.2)$$

Where, $M_p$ is the embedding dimension and $L_p$ is the time delay (or lag).

The techniques for identification of the optimal lag and the optimal embedding dimension are described below.

### 4.2.1.1 Identification of Lag using Mutual Information

In probability theory, the mutual information of two random variables is a measure of the variables' mutual dependence. The most common unit of measurement of mutual information is a bit. In the present study, mutual information is used to identify the nonlinear autocorrelation in the given time series $Y$. The mutual information at lag $l$ is denoted by

$$I(l) = \sum_{i=1}^{N-L} \sum_{j=L+1}^{N} p(y(i),y(j)) \log_2 \left( \frac{p(y(i),y(j))}{p(y(i),p(y(j))} \right) \quad (4.3)$$

Where $p(y(i),y(j))$ is the joint probability distribution function of $y(i)$ and $y(j)$, and $p(y(i))$ and $p(y(j))$ are the marginal probability distribution functions of $y(i)$ and $y(j)$ respectively.

The values of $I(l)$ are calculated for lags $l=1,2,\ldots$ and so on till, for some value of $l = L_p$, the first minimum value of $I$ is obtained. This $L_p$ is the optimal lag (or delay). The values of lag for the time series considered are presented in Table 4.2.
4.2.1.2 Identification of Embedding Dimension using False Nearest Neighbors

A method called the False Nearest Neighbor (FNN) was proposed in (Kennel, Brown and Abarbanel, 1992) to determine the minimal satisfactory embedding dimension. The idea is that for every point \( y(i) \) in the time series, we have to search for its nearest neighbor \( y(j) \) in an \( M_p \) dimensional space. Find the value of \( R_i \) which is given as

\[
R_i = \frac{|y(i+1)-y(j+1)|}{\|y(i)-y(j)\|}
\]  

(4.4)

If \( R_i \) exceeds a set threshold \( R_t \), the point \( y(i) \) is labeled as having a false neighbor. The criterion for finding the optimal \( M_p \) is that the percentage of false nearest neighbors drops to zero or very close to zero. The values of \( M_p \) obtained for time series considered are presented in Table 4.2.

<table>
<thead>
<tr>
<th>Stock</th>
<th>Lag</th>
<th>Embedding Dimension ( (M_p) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cipla</td>
<td>10</td>
<td>7</td>
</tr>
<tr>
<td>HUL</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>RIL</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>GSK</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>RBS</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>SBY</td>
<td>8</td>
<td>10</td>
</tr>
<tr>
<td>Boeing</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>3M</td>
<td>13</td>
<td>10</td>
</tr>
<tr>
<td>Yahoo</td>
<td>8</td>
<td>8</td>
</tr>
</tbody>
</table>

4.2.2 Clustering

Once the phase space representation of the time series is obtained, each row of the matrix \( Y_E \) is now considered to be a \( l \times M_p \) dimensional point. Hence, \( Y_E \) now has \( N-(M_p-I)L_p \) such points. These points are now clustered into groups which exhibit similarity in some sense, as defined by the clustering algorithms employed. In the present study, three different clustering techniques are evaluated, namely the UPGMA clustering, the k-means clustering and the FCM clustering algorithms.

4.2.2.1 k-Means Clustering

The k-Means technique (Han and Kamber, 2006) is one of the most effective techniques for clustering of data. The technique groups a given data set into \( k \) number of clusters \( (k \) is fixed a
Initially we select $k$ points arbitrarily from the dataset and assign them as centroids of the $k$ clusters. The next step is to take each of the remaining points from the data set and check its similarity to the $k$ centroids. The centroid of the cluster to which the point is most similar is the cluster to which that point belongs. When all the points have been assigned to any one of the $k$ clusters, the first step is completed. Next, the location of the $k$ centroids is updated based on the points in each cluster. After these, $k$ new centroids are obtained; the similarity of points in the dataset with each new cluster centroids is checked. The process repeats till the location of centroids stop changing. In the present study, the maximum number of iterations allowable is set as 1000. The similarity measure employed in the present study is the Cityblock distance measure, ie. the clustering algorithm will attempt to find

$$\min \left\{ \sum_{i=1}^{k} \sum_{O \in G_i} |O - C_i| \right\} \quad (4.5)$$

where, $k$ is the number of clusters

- $O$ is a point in the dataset
- $C_i$ is the location of the centroid of the $i$-th cluster
- $G_i$ is the $i$-th group or cluster

### 4.2.2.2 UPGMA Clustering

UPGMA clustering technique (Gose, Johnsonbaugh and Jost, 1996) is an agglomerative clustering technique where in initially, each point in the dataset is assumed to be belonging to its own cluster, ie. if there are $N$ points in the dataset then at the beginning of the clustering process, these are assumed to be $N$ clusters. The clusters are then iteratively merged together based on some measure of similarity to form larger and larger clusters till a user defined stopping condition is met or all the points become a part of the same cluster. The degree of similarity between two clusters say, $G_1$ and $G_2$ as employed by UPGMA is the mean of the all the distances between pairs of points, say $O_1$ in $G_1$ and $O_2$ in $G_2$. In the present study the Euclidean distance is employed and hence, the similarity measure $S$ is denoted by:

$$S = \frac{1}{|G_1||G_2|} \sum_{o_1 \in G_1} \sum_{o_2 \in G_2} \|O_1 - O_2\| \quad (4.6)$$
4.2.2.3 FCM Clustering (Bezdek, Ehrlich and Full, 1984)

Fuzzy clustering is a class of algorithms for cluster analysis in which the allocation of data points to clusters is not hard but "fuzzy", i.e. points in the dataset can belong to more than one cluster, and the degree to which a point belongs to a particular cluster is represented by its membership function. Assuming that the dataset consisting of \( N \) points has to be clustered into \( k \)-clusters, the membership value of each point \( O_i \) with respect to the cluster centroid \( C_j \), represented by \( \mu_{ij} \) is updated in each iteration based on the formula:

\[
\mu_{ij} = \frac{1}{\sum_{p=1}^{k} \left( \frac{\|o_i - c_j\|}{\|o_i - c_p\|} \right)^{2/(m-1)}} \tag{4.7}
\]

Where \( C_j \) is the centroid of the j-th cluster.

Similarly, the cluster centroid locations are also updated in each iteration based on the formula:

\[
C_p = \frac{\sum_{i=1}^{N} \mu_{ip}^m o_i}{\sum_{i=1}^{N} \mu_{ip}^m} , p = 1,2,\ldots,k \tag{4.8}
\]

The objective function of the FCM clustering algorithm is then,

\[
min \left\{ \sum_{i=1}^{N} \sum_{p=1}^{k} \mu_{ij}^m \|O_i - C_p\|^2 \right\} \tag{4.9}
\]

The algorithm iterates for the specified number of times or till either the minimum of the objective function is reached or the changes in the membership function matrix are within the specified tolerance levels. In the present study, the value of \( m=2 \), maximum number of iterations = 100 and a point is assigned to the cluster with the maximum membership value.

4.2.3 Identification of optimal number of clusters

Silhouette, as proposed in (Rousseeuw, 1987) is a method of interpretation and validation of clusters formed using clustering algorithms. In the present study, the average value of silhouettes is used to identify the optimal number of clusters obtained using the three different clustering algorithms considered. The technique provides a precise graphical representation of how well each point lies within its cluster.
Considering that there are \( k \) clusters, a point \( O_i \) belonging to cluster \( G_p \) has its dissimilarity to other points in the same cluster represented by

\[
d(O_i, G_p) = \frac{1}{|G_p|} \sum_{O_h \in G_p, O_h \neq O_i} \|O_i - O_h\| \tag{4.10}
\]

The dissimilarity of \( O_i \) with respect to points in another cluster, say \( G_s \) is given by

\[
d(O_i, G_s) = \frac{1}{|G_s|} \sum_{O_s \in G_s} \|O_i - O_s\| \tag{4.11}
\]

Consider

\[
D(O_i) = \{d(O_i, G_1), d(O_i, G_2), \ldots, d(O_i, G_k)\} \tag{4.12}
\]

Then the neighboring cluster of the cluster to which \( O_i \) belongs, is

\[
\text{neighbor} (O_i) = \min \{D(O_i) - d(O_i, G_p)\} \tag{4.13}
\]

The parameter \( S(O_i) \) is then defined as follows:

\[
S(O_i) = \frac{\text{neighbor} (O_i) - d(O_i, G_p)}{\max \{\text{neighbor} (O_i), d(O_i, G_p)\}} \tag{4.14}
\]

As can be seen from the above equation (4.14), an \( S(O_i) \) value close to 1 indicates that the point has been correctly clustered in \( G_p \). A value close to -1 indicates that the point is more suited for being clustered in the neighboring cluster. The average value of \( S(O_i) \) over all the points in the dataset is a measure of how well the points have been clustered. Plotting \( S(O_i) \) values for the clusters together in a figure is the silhouette plot and gives an idea of the quality of clustering. Narrower silhouettes indicate poor choice of the number of clusters. An example silhouette plot of RBS in-sample dataset is given in Figure 4.2.
4.2.4 Identifying profitable clusters

Once all the \( N-(M_p-1)L_p \) number of points have been clustered into, say \( k \) number of clusters (with \( k \) determined from the silhouettes), the centroids of these \( k \) clusters are analyzed to identify the ‘profitable’ clusters. The process is described below.

Assume that the centroid for the \( i \)-th cluster to be

\[
Y_{\text{Centroid},i} = [y_{i,M_p} \ldots y_{i,2} \ y_{i,1}]
\]

Where

The cluster \( i \) is defined to be profitable if its centroid \( Y_{\text{Centroid},i} \) satisfies the criterion:

\[
100 \left( \frac{y_{i,1} - y_{i,2}}{y_{i,2}} \right) > 0.5
\]

Otherwise, the cluster is considered to be non-profitable.

A value of 0.5 is chosen to ensure that the gains are high enough (at least 0.5%) to compensate for the transaction costs.

All the points in all the profitable clusters taken together (say, a total of \( H \) number of points), where \( H < N-(M_p-1)L_p \), are considered for obtaining the training input set. In order to forecast the
possibility of trading, at the latest point in time, say ‘t’, the latest $I \times M_p$ vector is considered to be:

$$Y(t) = \begin{bmatrix} y(t - (M_p - 2) L_p) & \ldots & y(t) & y(t + L_p) \end{bmatrix} \quad (4.17)$$

The aim of the trading recommender is to take in the incomplete $(M_p - 1)$ element vector and attempt to generate $L_p$ days-ahead trading recommendations. For the purpose, all the points in the profitable clusters are truncated by removing the latest samples, ie. the truncated vector $Y_r$ would be now:

$$Y_r(t) = \begin{bmatrix} y(t - (M_p - 2) L_p) & \ldots & y(t) \end{bmatrix} \quad (4.18)$$

The resulting $H$ points with $(M_p - 1)$ dimensions is used as the training input to the ARTMAP Network.

4.2.5 Generating trading recommendations using ARTMAP

The ARTMAP network, also called Predictive ART, as proposed in (Carpenter, Grossberg and Reynolds, 1991) is an extension of the ART network and unlike ART, is a supervised network. The ARTMAP network offers significant advantages over conventional back propagation neural networks as they are self-organizing and self-stabilizing. Block diagram representation of the ARTMAP network is presented in Figure 4.3.
The ARTMAP network uses two ART networks with the first network $\text{ART}_a$ taking the input feature vector as the input and the second network $\text{ART}_b$ takes the corresponding target output as the input. Each node in the $F_2^a$ field represents a recognition category of patterns given as input to $\text{ART}_a$. Each such node is associated to an $\text{ART}_b$ category in the $F_2^b$ field through the inter-$\text{ART}$ map field.

In the present study, the input features are complementary coded. After complementary coding, the input dimension becomes $2^*(M_p-1)$ and hence the training set is a $2^*(M_p-1) \times H$ matrix. There are two classes: ‘Trade’, denoted by +1 and ‘No Trade’ denoted by -1. The vigilance parameter is set to 0.75 for all datasets and the learning rate is set to 1.

Based on the clustering technique used (k-means, UPGMA or the FCM), three variants of the proposed system are designed and evaluated. The results are presented in the Appendix B in tables B.1-B.9.

4.3 Other stock trading recommenders considered

In addition to the proposed system, three other trading recommender systems described below, are also evaluated.
4.3.1 Phase space-clustering based recommender system

This recommender system employs the state space representation of the stock price time series along with clustering techniques for generating stock trading recommendations. Once the time series, as in equation (4.1) is transformed to its phase space representation, as in equation (4.2), the points are clustered into similar groups using clustering techniques. Three clustering techniques, namely the k-means, UPGMA and FCM (described in section 4.2.2) are considered resulting three different variants of the clustering based recommender system. The optimal number of clusters \((K)\) is identified using silhouettes (described in section 4.2.3). Profitable clusters are then identified using the technique described in section 4.2.4. Assuming that a total of \(P\) profitable and \(Q\) non-profitable clusters are identified such that \(P+Q=K\). The centroid of cluster \(i\), where \(1 \leq i \leq K\) is represented as

\[
C_i = \begin{bmatrix} y_{c,i}(1) & \ldots & y_{c,i}(L_p) & y_{c,i}(1) \end{bmatrix} \tag{4.19}
\]

The centroids are then truncated by removing the latest sample to yield \(M-1\) dimensional centroids. The truncated centroid of cluster \(i\), where \(1 \leq i \leq K\) is now represented as

\[
C_{Ti} = \begin{bmatrix} y_{c,i}(1) & \ldots & y_{c,i}(L_p) \end{bmatrix} \tag{4.20}
\]

At time \(\tau\), the phase space representation vector, as given in equation (4.17) is

\[
Y(\tau) = \begin{bmatrix} y(\tau - (M_p - 2)L_p) & \ldots & y(\tau) & y(\tau + L_p) \end{bmatrix} \tag{4.21}
\]

Where \(y(\tau)\) is the stock price at instant \(\tau\). Now assuming that the stock price at instant \(\tau\) is \(y(\tau)\), the \(L_p\) day-ahead value is \(y(\tau+L_p)\). However, since the stock price \(L_p\) steps into future is unknown, only the first \(M_p-1\) samples are available. Hence the \(Y(\tau)\) vector is truncated to

\[
Y_r(\tau) = \begin{bmatrix} y(\tau - (M_p - 2)L_p) & \ldots & y(\tau) \end{bmatrix} \tag{4.22}
\]
It is considered that the cluster $i$ with the minimum $\|C_i - Y(\tau)\|$ is the cluster with minimum $\|C_{Yi} - Y_r(\tau)\|$ value, as well. This observation was found to be true during the in-sample testing for all the stocks. Hence, if $C_i$ represents the centroid of a profitable cluster, then stock is bought on the day’s closing price and sold $L_p$ days afterwards. On the other hand, if $C_i$ represents the centroid of a non-profitable cluster, no trade is initiated.

4.3.2 Time series-ARTMAP based recommender system

Time series-ARTMAP based recommender system is the most basic implementation of the recommender system considered in the present study. The ARTMAP network is trained to generate one-day-ahead trading recommendations. Consider the stock price time series consisting of $N$ samples as given in equation (4.1). The supervisory signal is +1 (Trade) if $y(i) > y(i-1)$ and -1 (No Trade) if $y(i) < y(i-1)$, where $1 \leq i \leq N$.

Once the system is trained, the stock price on day $\tau$, $y(\tau)$ is input to the system and the appropriate trading recommendation (Trade/No Trade) for the day $\tau+1$ is generated. If the recommendation is ‘Trade’ then the stock is bought on the day’s closing and sold on the next day’s close. A transaction cost of 0.5% is considered.

4.3.3 Phase space representation- ARTMAP based recommender system

This variant of the recommender system first identifies the phase space representation of the given stock price time series as given in equation (4.2). However, there is no identification of the ‘profitable’ clusters. The supervisory signal $S_y$, for the ARTMAP at instant $\tau$ is generated based on the condition:

$$S_y(\tau) = +1 \text{ if } \left( \frac{y(\tau)-y(\tau-L_p)}{y(\tau-L_p)} \right) > 0$$

$$-1 \text{ otherwise} \quad (4.23)$$

As in the proposed recommender system, for the training process, the $M_p$ dimensional vector is truncated to an $M_p-I$ dimensional vector by removing the latest sample, as in equation (4.18).
4.4 Results

All the four types of clustering based recommender systems are evaluated, namely, time series-ARTMAP based recommender, phase space –ARTMAP based recommender, phase space – clustering based recommender and the phase space – clustering- ARTMAP based recommender. Three variants each, of phase space – clustering based recommender and the phase space – clustering- ARTMAP based recommender systems, based on the clustering technique employed (k-means, UPGMA and FCM), are evaluated. Hence a total of eight different recommender systems are evaluated and the results are presented in Appendix B, in tables B.1-B.9. Performance of each of the recommender systems is evaluated on eight different performance measures as suggested in (Brabazon and O’Neill, 2006), namely, TP, AP, P/ST, L/LT, MD, TT, PF and WR. Results are compared to B&H returns over the same time frame. The overall in-sample and out-sample profits generated by all the recommender systems are summarized in figures 4.4-4.9 below. The in-sample profits are presented in figures 4.4-4.6.

**Figure 4.4 Clustering based recommender systems in-sample profits -Indian stocks**

**Figure 4.5 Clustering based recommender systems in-sample profits -UK stocks**
Figure 4.6 Clustering based recommender systems in-sample profits - US stocks

The out-sample profits generated by all the recommender systems for all the nine stocks considered in the present study is presented in figures 4.7-4.9.

Figure 4.7 Clustering based recommender systems out-sample profits-Indian stocks

Figure 4.8 Clustering based recommender systems out-sample profits-UK stocks
4.5 Conclusions

It can be clearly observed from tables B.1-B.9 in Appendix B and the figures 4.7-4.9, that the phase-space-clustering-ARTMAP based recommender systems tend to significantly outperform the traditional B&H strategy. The second best performance was demonstrated by phase space-clustering based recommender systems. It is also observed that of the three variants of the phase space-clustering-ARTMAP based recommenders, the phase space-k means clustering-ARTMAP based recommender system tends to generate higher profits than the other two variants. From the results, the effectiveness of clustering based stock trading recommender systems in generating excess returns from stock markets has been empirically verified.